

This Page Is Inserted by IFW Operations  
and is not a part of the Official Record

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning documents *will not* correct images,  
please do not report the images to the  
Image Problems Mailbox.**

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 09-135177

(43)Date of publication of application : 20.05.1997

(51)Int.Cl.

H03M 13/12

**H04L 1/00**

(21)Application number : 07-288700

(71)Applicant : TOSHIBA CORP

(22)Date of filing : 07.11.1995

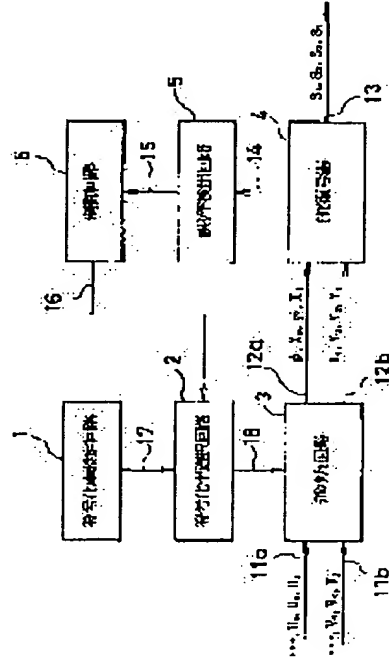
(72)Inventor : AIZAWA MASAMI  
OKITA SHIGERU

**(54) METHOD FOR JUDGING VITERBI DECODING SYNCHRONIZATION AND ITS DEVICE**

**(57)Abstract:**

**PROBLEM TO BE SOLVED:** To provide a viterbi decoding synchronization judging method and its device capable of detecting a correct encoding rate quickly by executing depuncture only conforming the encoding rate which can be selected by the rule of a transmitting system.

**SOLUTION:** The plural encoding rates are previously set by an encoding rate setting circuit from an outside, the optional encoding rate is selected by an encoding selecting circuit 2 from the set encoding rate, a code is made-up by a depuncture circuit 3 based on the selected encoding rate and a viterbi decoding is executed by a viterbi decoder 4. The error rate of a received code during a prescribed period is obtained by an error rate detecting circuit 5, it is judged whether or not code making-up is correct by a control circuit 6 based on the error rate, and when not correct, the different encoding rate is newly selected by an encoding rate selecting circuit 2 and control is executed so that code making-up, viterbi decoding and error rate judgement are repeated through the use of the new encoding rate.



## LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平9-135177

(43) 公開日 平成9年(1997)5月20日

(51) Int.Cl. <sup>a</sup>	識別記号	庁内整理番号	F I	技術表示箇所
H 0 3 M 13/12			H 0 3 M 13/12	
H 0 4 L 1/00			H 0 4 L 1/00	A C

審査請求 未請求 請求項の数 5 O L (全 15 頁)

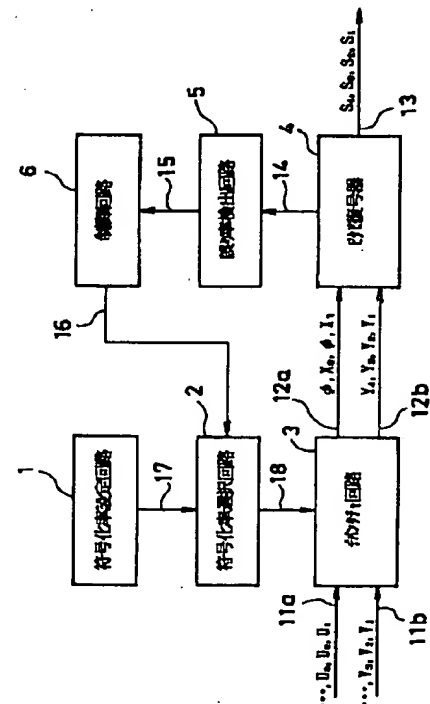
(21) 出願番号	特願平7-288700	(71) 出願人	000003078 株式会社東芝 神奈川県川崎市幸区堀川町72番地
(22) 出願日	平成7年(1995)11月7日	(72) 発明者	相沢 雅己 神奈川県横浜市磯子区新杉田町8番地 株 式会社東芝マルチメディア技術研究所内
		(72) 発明者	沖田 茂 神奈川県横浜市磯子区新杉田町8番地 株 式会社東芝マルチメディア技術研究所内
		(74) 代理人	弁理士 三好 秀和 (外3名)

(54) 【発明の名称】 ビタビ復号同期判定方法および装置

(57) 【要約】

【課題】 伝送方式の規定により選択され得る符号化率についてのみデバଙ୍କチャを行い、正しい符号化率をより早く検出することが可能なビタビ復号同期判定方法およびその装置を提供する。

【解決手段】 符号化率設定回路1により外部から予め複数の符号化率を設定し、設定された符号化率から任意の符号化率を符号化率選択回路2により選択し、選択された符号化率に基づいてデバଙ୍କチャ回路3により符号補填して、ビタビ復号器4によりビタビ復号を施す。受信符号の所定期間中の誤り率を誤り率検出回路5により求め、制御回路6により誤り率に基づいて符号補填が正しいか否かを判定し、正しくない場合に符号化率選択回路2に異なる符号化率を新たに選択せしめ、この新たな符号化率を用いて符号補填、ビタビ復号及び誤り率判定を繰り返すように制御する。



## 【特許請求の範囲】

【請求項 1】 予め設定された複数の符号化率から状況に応じて選択された符号化率となるようにたたみ込み符号に対して符号削除を施して送信された信号を受信して復号するビタビ復号同期判定方法において、  
予め複数の符号化率を設定する第 1 の過程と、  
前記複数の符号化率から第 1 の符号化率を選択する第 2 の過程と、  
該選択された符号化率に基づいて受信信号に対して符号補填を施す第 3 の過程と、  
該符号補填された信号に対してビタビ復号を施す第 4 の過程と、  
所定期間中の誤り検出に基づいて誤り率を求める第 5 の過程と、  
求められた誤り率に基づいて前記符号補填が正しいか否かを判定する第 6 の過程と、  
符号補填が正しくないと判定される場合に、前記複数の符号化率から前記第 1 の符号化率とは異なる第 2 の符号化率を選択して、前記第 3 の過程から前記第 6 の過程を繰り返す第 7 の過程と、  
を含むことを特徴とするビタビ復号同期判定方法。

【請求項 2】 前記第 5 の過程における誤り検出には、ビタビ復号の最尤パスを使用することを特徴とする請求項 1 記載のビタビ復号同期判定方法。

【請求項 3】 前記第 5 の過程における誤り検出には、ビタビ復号のバスメトリックの正規化量又は正規化頻度を使用することを特徴とする請求項 1 記載のビタビ復号同期判定方法。

【請求項 4】 前記第 5 の過程における誤り検出には、受信信号を簡易復号した信号とビタビ復号した信号を再符号化した信号とを比較し、この比較結果に基づいて誤り検出することを特徴とする請求項 1 記載のビタビ復号同期判定方法。

【請求項 5】 複数の符号化率から状況に応じて選択された符号化率となるようにたたみ込み符号に対して符号削除を施して送信された信号を受信して復号するビタビ復号同期判定装置において、  
外部から予め複数の符号化率を設定する符号化率設定手段と、  
前記設定された符号化率から任意の符号化率を選択する符号化率選択手段と、  
該選択された符号化率に基づいて受信信号に対して符号補填を施す符号補填回路と、  
符号補填された信号に対してビタビ復号を施すビタビ復号器と、  
所定期間中の誤り検出に基づいて誤り率を求める誤り率検出手段と、  
求められた誤り率に基づいて前記符号補填が正しいか否かを判定し、符号補填が正しくないと判定される場合に、前記符号化率選択手段に前記選択された符号化率と

は異なる符号化率を選択せしめ、新たに選択された符号化率を用いて符号補填、ビタビ復号及び誤り率判定を繰り返す制御を行う制御手段と、  
を備えることを特徴とするビタビ復号同期判定装置。

## 【発明の詳細な説明】

## 【0001】

【発明の属する技術分野】 本発明はたたみ込み符号化—ビタビ復号法に係り、特に符号間引き（パンクチャ）された信号を受信して復号するビタビ復号同期判定方法およびその装置に関する。

## 【0002】

【従来の技術】 デジタル伝送における誤り訂正法として、ビタビ復号法（G.D. Forney, Jr., "The Viterbi Algorithm" Proceedings of IEEE, vol.61, pp 268-278, Mar. 1973. 参照）がある。このビタビ復号法は、たたみ込み符号の最尤復号を効率よく実現するアルゴリズムである。

【0003】 最初に、このビタビ復号法について概説する。ビタビ復号法が適用されるたたみ込み符号器は、例えば図 6 に示すようなたたみ込み符号器を用いて送信側で符号化（この例では、符号化率  $R = 1/2$ 、拘束長  $L = 3$ ）が施される。図 6 の符号器は、2 ビットのシフトレジスタと 2 回路の排他的論理和回路からなり、符号器の内部状態（ $a$ 、 $b$ ）は、4 通りの状態をとることができる。そして、1 ビットの入力信号  $s_i$  当たり 2 ビットの符号化出力信号  $X_i$ 、 $Y_i$  が得られ（符号化率  $R = 1/2$ ）、入力の 1 ビットの変化は連続する出力 3 ビットに影響する（拘束長  $L = 3$ ）。

【0004】 次に、受信側での復号動作を図 7 に示すトレリス表現に基づいて説明する。各時刻では符号器の内部状態（ $a$ 、 $b$ ）に対応する 4 つの状態が存在し、1 シンボル受信するごとに、状態遷移していく。ここで各状態において受信信号との距離を表す値としてバスメトリック値があり、一つ前の状態のバスメトリックと受信信号からなるブランチメトリックとの和のより小さい方を新しいバスメトリックとして選択していき、そのうち最もバスメトリックの小さい状態をさかのぼっていったパス（ある程度の長さで実際には収束する）を最も確からしいパスすなわち最尤パスとして復号（誤り訂正）を行う復号法がビタビ復号法である。

【0005】 この図 7 を参照するに、各太線は時刻  $k = 7$  まで復号をすすめたときの、各状態（ $a$ 、 $b$ ）＝

{0, 0}、{0, 1}、{1, 0} 及び {1, 1} においてそれぞれ選択され生き残った生き残りパス（復号系列の候補） $V_{0,7}$ 、 $V_{1,7}$ 、 $V_{2,7}$  及び  $V_{3,7}$  を表すものである。この生き残りパスは、受信符号系列と伝送符号系列のハミング距離差（以下、ハミング距離差を単に距離差と略す）をもとに選択される。各時刻まで復号をすすめたときの、その距離差に相当するバスメトリックを図 7 では実線の四角で表している。点線の四角は捨

てられたバスのバスメトリックである。

【0006】図7から明らかなように、時刻： $k=7$ まで復号をすすめたとときの生き残りバス $V_{0,7}$ 、 $V_{1,7}$ 、 $V_{2,7}$ 及び $V_{3,7}$ のバスメトリックは、それぞれ3、3、2、2である。一般に、各生き残りバスの過去の系列ほど1本にまとまる確率が高いので、生き残りバスのメモリ長を適当な長さ（拘束長の4～6倍）で打ち切り、最過去のシンボルをその時刻の復号シンボルとして出力する。

【0007】また誤りパターンによっては各生き残りバスの最過去のビットが一致しないことがままあるが、上記バスメトリックが最小のものが、最も確からしい復号系列に相当することはいうまでもない。

【0008】ビタビ復号の装置化において、バスメトリックの演算は図8に示す状態遷移の組を単位として実現できる。時刻 $(k-1)$ で選択された生き残りのバスメトリックをそれぞれ $\Gamma_{k-1}$ 、 $\Gamma'_{k-1}$ とし、現在の受信符号と可能な送信符号との距離差に相当するブランチメトリックを $\lambda_k$ 、 $\lambda'_k$ とする。現在の時刻 $k$ における生き残りバスの候補は、各状態で2つつ存在し、それぞれのバスメトリックは $\Gamma_{k-1}$ 、 $\Gamma'_{k-1}$ 、 $\lambda_k$ 、 $\lambda'_k$ を用いて $(\Gamma_{k-1} + \lambda_k)$ 、 $(\Gamma'_{k-1} + \lambda'_k)$ 、 $(\Gamma_{k-1} + \lambda'_k)$ 、 $(\Gamma'_{k-1} + \lambda_k)$ で表される。また各状態ではそれぞれバスメトリックの内、小さい方に相当するバスが選択される。

【0009】このようにバスメトリックの演算は、加算(Add)、比較(Compare)及び選択(Select)の操作で実現できるので、このようなバスメトリックの演算器をACSユニット(ACSU)と呼ばれている。

【0010】図9のブロック図は、拘束長 $L=7$ のビタビ復号器の構成例を示すものである。図9において、ブランチメトリックユニットBMU901は現在の受信符号と可能な出力符号との距離差を計算してブランチメトリックを出力するユニットである。ACSU904の数は、可能な状態数を $N_s = 2^{L-1}$  ( $L$ : 拘束長)とすると、それぞれが状態2個分に対応するので、 $(N_s/2)$ 個である。したがってこの従来例では、拘束長 $L=7$ であるからたたみ込み符号器の状態数は $N_s = 2^{L-1} = 64$ となり、最尤判定部905の比較入力の数も64となる。

【0011】比較選択回路により構成される最尤判定部905は、最も確からしい生き残りバス（最尤バス）を判定するために、最小のバスメトリックを検出することを目的とするものである。

【0012】バスメトリック更新回路907は各状態で残すバスを更新することを目的とするものである。すなわち、各状態 $\{0, 0, 0, 0, 0, 0\} = (0)$ 、 $\{0, 0, 0, 0, 0, 1\} = (1)$ 、 $\{0, 0, 0, 0, 1, 0\} = (2)$ 、…、 $\{1, 1, 1, 1, 1, 1\}$ 、

$1\} = (63)$ 、で選択し残したバスを示すバス選択信号 $\beta(0)$ 、 $\beta(1)$ 、 $\beta(2)$ 、…、 $\beta(63)$ 、により $V(0)$ 、…、 $V(63)$ の最過去シンボルに相当する復号シンボルの候補 $\sigma(0)$ 、…、 $\sigma(63)$ を出力する。

【0013】ビタビ復号のセクタ908はこれらの復号シンボルの候補 $\sigma(0)$ 、…、 $\sigma(63)$ から最尤バスに相当する復号シンボルを選択し、ビタビ復号シンボルとして出力するものである。この選択には最尤判定部905から出力される最尤バスを示す識別信号 $P_m$  ( $m=0$  or  $1$  or  $2$  … or  $63$ )を用いる。

【0014】ところで、バスメトリックの値 $\Gamma(0)$ 、…、 $\Gamma(63)$ は、このままブランチメトリックを累積していくと、限りなく大きくなっていく。実際の装置化においては、ACSユニット内のバスメトリックを保持するバスメトリックレジスタ、のサイズは有限のため、時間がたつとオーバーフローを起こすことになる。

【0015】このオーバーフローを防ぐためには、最小バスメトリックで正規化すれば良い。すなわち、バスメトリックレジスタに保存する前に、最尤バスメトリックレジスタに保存されている1単位時刻前の最小バスメトリック $\Gamma_{min, k-1}$ で各バスメトリックを減算する。これは図9に示すように、予めBMU901で計算したブランチメトリック $\lambda_{00}$ 、 $\lambda_{01}$ 、 $\lambda_{10}$ 、 $\lambda_{11}$ 、から正規化回路902により、最小バスメトリック $\Gamma_{min, k-1}$ を減算しても同じことである。こうすることで各状態に残されるバスメトリック $\Gamma(0)$ 、…、 $\Gamma(63)$ は、ある範囲に収まることになり、またバスメトリックレジスタのサイズを十分大きくとっておけば、復号性能に影響は生じない。

【0016】ところで、上記のようなたたみ込み演算によりたたみ込み符号化を行うと、冗長性が高まるが、その分伝送すべき符号系列の長さが増加することになる。たたみ込み符号の受信系列において、幾つかの符号誤りが生じて（あるいは符号欠落が生じて）も、適当に埋めることにより復号が可能である。すなわち、幾つかの符号を削除（バンクチャ）して伝送することにより、たたみ込み符号の冗長度をある程度低下させても復号が可能であり、このようなバンクチャド符号を使うことである程度の任意の符号化率を選択することができる。

【0017】こうしてバンクチャド符号を用いると、伝送路の状況が悪いときは符号化率が低く誤り訂正能力の高い符号を用い、伝送路の状況が良いときは誤り訂正能力は低い符号化率の高い符号を用いるというように、伝送効率と誤り訂正能力とのトレードオフが成立し、伝送路の状況に応じた可変符号化率符号が実現できる。

【0018】次に、バンクチャド符号における符号間引きバンクチャの様子を図10に示す。入力信号系列101として $S_1$ 、 $S_2$ 、 $S_3$ 、 $S_4$ が入力されると、たたみ込み符号器102により、符号化系列103が $\{X$

1, Y1), {X2, Y2}, {X3, Y3}, {X4, Y4}として出力される。

【0019】バンクチャ回路104は、符号化系列103のうち、X2, X4を削除し(伝送しない)送信系列105, {U1(X1), V1(Y1)}, {U2(Y2), V2(X3)}, {U3(Y3), V3(Y4)}, ...を出力する。これにより符号化率 $R=4/6=2/3$ となる。符号化率指定回路107は、可能な複数の符号化率から伝送路の状況等により選択された符号化率に応じて、バンクチャ回路104に符号化率または

10 バンクチャパターンを指示するためのものである。  
【0020】ここで例えば伝送方式として、直交符号化方式(QPSK)等を使用した場合、1シンボルで2ビットの情報が送ることができる。受信側では、バンクチャされた信号がシーケンシャルに伝送され、どの符号をバンクチャ(削った)のかは判明していない。 $R=4/*$

\* $6=2/3$ の場合、6ビットを1周期で伝送され、1シンボルあたり2ビット伝送するとして、3シンボル周期で情報が伝達される。この場合、送信側のバンクチャパターンが一定であっても、受信側で削除された符号位置を埋めるデバンクチャ可能なパターンは3シンボル周期に対応して3通りあり、自由度は3になる。

【0021】そこで一定期間の誤り状態を判定し、定めた誤り率を上回る場合には、1シンボルずつデバンクチャ位置をずらし、再度、誤り状態を判定する。この場合は3つのバンクチャ位置について、誤り率を調べて、すべての誤り状態が悪いと判定されれば、符号化率が間違っていたと判断し、異なる符号化率を調べる。符号化率により、それぞれ異なる自由度があり、表1に各符号化率ごとにその自由度を示す。

【0022】

【表1】

符号化率	バンクチャ・パターン	マッピング	自由度
1/2	X: 1 Y: 1	I = X1 Q = Y1	1
2/3	X: 1010 Y: 1111	I = X1 Y2 Y3 Q = Y1 X3 Y4	3
3/4	X: 101 Y: 110	I = X1 Y2 Q = Y1 X3	2
4/5	X: 1000 1000 Y: 1111 1111	I = X1 Y2 Y4 Y5 Y7 Q = Y1 Y3 X5 Y6 Y8	5
5/8	X: 10101 Y: 11010	I = X1 Y2 Y4 Q = Y1 X3 X5	3
6/7	X: 100101 100101 Y: 111010 111010	I = X1 Y2 X4 X6 Y7 Y9 Y11 Q = Y1 Y3 Y5 X7 Y8 X10 X12	7
7/8	X: 1000101 Y: 1111010	I = X1 Y2 Y4 Y6 Q = Y1 Y3 X5 X7	4
8/8	X: 10001011 10001011 Y: 11110100 11110100	I = X1 Y2 Y4 Y6 X8 Y9 Y11 X13 X15 Q = Y1 Y3 X5 X7 X9 Y10 Y12 Y14 X16	8

また、送信側では伝送状況に応じて、あるいは階層伝送などにより、異なる符号化率、バンクチャで伝送を行う場合があり、その点を考慮して、受信側では使用されるすべての符号化率に対応するように構成する必要がある。表1に示したように、すべての符号化率1/2~8/9における自由度の合計は34となり、すべての符号化率に対応すると最大34回誤り率を判定する必要がある。しかもビタビ復号は前に述べた様に、パスが収束するまでに時間を要するため、誤りの判定にはある程度の時間を必要とし、複数の符号化率、バンクチャ位置判定には長い時間がかかる。

【0023】このようなバンクチャド符号を復号する集積回路等を実際に作成する場合には、さまざまな伝送方式に対応可能なように汎用性を考慮して、多くの符号化率に対応できる構成をとる。しかし伝送方式の規定によっては、実際に使用される符号化率はそのうち一部の符号化率に限定されることがある。この場合でも、バンクチャド符号の復号器に上記のように作成された汎用の集積回路を用いていたので、送信側が選択した符号化率を見つ

ド符号の復号に多くの時間を費やしていた。

【0024】従来のバンクチャド符号を復号する集積回路における符号化率を検索するフローチャートを図11に示す。同図において、まず符号化率を $R=1/2$ と仮定してバンクチャ位置検出を行う(ステップS110)。次いで誤り率が所定の誤り率より小さいか否かを判定する(ステップS115)。誤り率が所定の誤り率より小さければ(ステップS115の判定でYes)、この仮定した符号化率 $R=1/2$ が正しいものとして同期判定へ移る。誤り率が所定の誤り率に等しいか大きければ(ステップS115の判定でNo)、この仮定した符号化率 $R=1/2$ が誤りであるので、次に符号化率を $R=2/3$ と仮定してバンクチャ位置検出を行う(ステップS120)。

【0025】次いで誤り率が所定の誤り率より小さいか否かを判定する(ステップS125)。誤り率が所定の誤り率より小さければ(ステップS125の判定でYes)、この仮定した符号化率 $R=2/3$ が正しいものとして同期判定へ移る。誤り率が所定の誤り率に等しいか大きければ(ステップS125の判定でNo)、この仮

定した符号化率  $R = 2/3$  が誤りであるので、次に符号化率を  $R = 3/4$  と仮定してバンクチャ位置検出を行う（ステップ S130）。次いで誤り率が所定の誤り率より小さいか否かを判定する（ステップ S135）。誤り率が所定の誤り率より小さければ（ステップ S135 の判定で Yes）、この仮定した符号化率  $R = 3/4$  が正しいものとして同期判定へ移る。誤り率が所定の誤り率に等しいか大きければ（ステップ S135 の判定で No）、この仮定した符号化率  $R = 3/4$  が誤りである。

【0026】以下、同様に表 1 に基づいて次々に符号化率を変えながら、誤り率が所定の誤り率より小さくなるまで繰り返す。そして、符号化率を  $R = 8/9$  と仮定したバンクチャ位置検出（ステップ S180）を行った後の誤り率判定（ステップ S185）において、誤り率が所定の誤り率より小さければ（ステップ S185 の判定で Yes）、この仮定した符号化率  $R = 8/9$  が正しいものとして同期判定へ移る。誤り率が所定の誤り率に等しいか大きければ（ステップ S185 の判定で No）、この仮定した符号化率  $R = 8/9$  が誤りであるので、最初の符号化率を再度試みるためにステップ S110 へ移る。

【0027】このように、従来のバンクチャド符号の復号器に上記のように作成された汎用の集積回路を用いた場合には、送信側が選択した符号化率を見つけるまでバンクチャド符号の復号に多くの時間を費やし、最悪すべての符号化率を検査するまで、最大 34 通りのバンクチャパターンを調べていた。

【0028】

【発明が解決しようとする課題】上記述べたように、さまざまな伝送方式に対応するように構成した復号器においては、伝送方式の規定により不要であるのが明確な符号化率まで、順次符号化率を変えながら調べていくために、正しい符号化率を見つけるまでに長い時間がかかるという問題点があった。

【0029】以上の問題点に鑑み、本発明の課題は、予め伝送方式の規定により選択され得る符号化率を指定し、受信側で考慮する必要のない符号化率に対しては復号を省略して、正しい符号化率をより早く検出することが可能なビタビ復号同期判定方法およびその装置を提供することである。

【0030】

【課題を解決するための手段】上記課題を解決するために、本発明は次の構成を有する。すなわち請求項 1 記載の発明は、予め設定された複数の符号化率から状況に応じて選択された符号化率となるようにたたみ込み符号に対して符号削除を施して送信された信号を受信して復号するビタビ復号同期判定方法において、予め複数の符号化率を設定する第 1 の過程と、前記複数の符号化率から第 1 の符号化率を選択する第 2 の過程と、該選択された符号化率に基づいて受信信号に対して符号補填を施す第

3 の過程と、該符号補填された信号に対してビタビ復号を施す第 4 の過程と、所定期間中の誤り検出に基づいて誤り率を求める第 5 の過程と、求められた誤り率に基づいて前記符号補填が正しいか否かを判定する第 6 の過程と、符号補填が正しくないと判定される場合に、前記複数の符号化率から前記第 1 の符号化率とは異なる第 2 の符号化率を選択して、前記第 3 の過程から前記第 6 の過程を繰り返す第 7 の過程と、を含むことを要旨とする。

【0031】また、請求項 2 記載の発明は、請求項 1 記載のビタビ復号同期判定方法において、前記第 5 の過程における誤り検出には、ビタビ復号の最尤パスを使用することを要旨とする。

【0032】また、請求項 3 記載の発明は、請求項 1 記載のビタビ復号同期判定方法において、前記第 5 の過程における誤り検出には、ビタビ復号のパスメトリックの正規化量又は正規化頻度を使用することを要旨とする。

【0033】また、請求項 4 記載の発明は、請求項 1 記載のビタビ復号同期判定方法において、前記第 5 の過程における誤り検出には、受信信号を簡易復号した信号とビタビ復号した信号を再符号化した信号とを比較し、この比較結果に基づいて誤り検出することを要旨とする。

【0034】また、請求項 5 記載の発明は、複数の符号化率から状況に応じて選択された符号化率となるようにたたみ込み符号に対して符号削除を施して送信された信号を受信して復号するビタビ復号同期判定装置において、外部から予め複数の符号化率を設定する符号化率設定手段と、前記設定された符号化率から任意の符号化率を選択する符号化率選択手段と、該選択された符号化率に基づいて受信信号に対して符号補填を施す符号補填回路と、符号補填された信号に対してビタビ復号を施すビタビ復号器と、所定期間中の誤り検出に基づいて誤り率を求める誤り率検出手段と、求められた誤り率に基づいて前記符号補填が正しいか否かを判定し、符号補填が正しくないと判定される場合に、前記符号化率選択手段に前記選択された符号化率とは異なる符号化率を選択せしめ、新たに選択された符号化率を用いて符号補填、ビタビ復号及び誤り率判定を繰り返す制御を行う制御手段と、を備えることを要旨とするビタビ復号同期判定装置である。

【0035】本発明に係るビタビ復号同期判定方法及びビタビ復号同期判定装置において、外部から予め複数の符号化率を設定する符号化率設定手段は、入力信号端子であってもよいし、システム立ち上げ時に設定されるレジスタであってもよい。また、それぞれの符号化率ごとに当該符号化率を使用されるか否かを示す信号を入力する直接指定により符号化率を設定してもよいし、各伝送システムにおいて使用される符号化率の組を予め複数組記憶し、いずれの符号化率の組が使用されるかを指定する間接指定により符号化率を設定してもよい。

【0036】本発明に係るビタビ復号同期判定方法及び

ビタビ復号同期判定装置によれば、予め設定された複数の符号化率についてのみ受信信号に符号補填を行い、予め設定されない符号化率については考慮する必要がないので、送信側が選択した符号化率を受信側が見つかるまでの時間を短縮することができる。

【0037】

【発明の実施の形態】次に図面を参照して、本発明の実施の形態について詳細に説明する。図1は、本発明に係るビタビ復号同期判定装置の第1の実施の形態を示すブロック図である。同図において、本発明に係るビタビ復号同期判定装置は、外部から予め複数の符号化率を設定する符号化率設定回路1と、設定された符号化率から任意の符号化率を選択する符号化率選択回路2と、選択された符号化率に基づいて受信信号に対して符号補填を施すデバଙ୍କチャ回路3と、符号補填された信号に対してビタビ復号を施すビタビ復号器4と、所定期間中の誤り検出に基づいて誤り率を求める誤り率検出回路5と、求められた誤り率に基づいて前記符号補填が正しいか否かを判定し、符号補填が正しくないと判定される場合に、符号化率選択回路2に前記選択された符号化率とは異なる符号化率を選択せしめ、新たに選択された符号化率を用いて符号補填、ビタビ復号及び誤り率判定を繰り返す制御を行う制御回路6とを備えて構成されている。

【0038】図2は、符号化率設定回路1の詳細な実施の形態を説明する詳細回路図である。図2(a)の符号化率設定回路は、それぞれの符号化率ごとに独立した入力信号端子を備えて、それぞれの端子の論理レベルを“1”または“0”にクランプすることにより、それぞれの符号化率が使用されるか使用されないかを指定する $R=1/2$ 設定信号～ $R=8/9$ 設定信号の8本の符号化率設定信号を生成する。

【0039】図2(b)の符号化率設定回路は、8ビットのシフトレジスタを用いた符号化率設定回路であり、シリアルデータ入力端子、クロック端子及び図2(a)と同様な8本の符号化率設定信号出力を備え、本装置の初期設定時に8ビットの符号化率設定信号がシリアルデータ入力端子から入力され、クロック信号により順次上位のフリップフロップにシフトされて設定される。

【0040】次に、本実施の形態のビタビ復号同期判定装置の動作を説明する。符号化率 $R=4/6=2/3$ 、1シンボルあたり2ビットの伝送を行う場合を例として説明する。他の符号化率においても、自由度が異なるだけで、ほぼ同様である。

【0041】送信側の符号化動作は、従来例と全く同様である。図10を参照して符号間引きであるバଙ୍କチャの様子を説明する。入力信号系列41としてs0, s1, s2, s3が入力されると、たたみ込み符号器42により符号化系列43が{X1, Y1}, {X2, Y2}, {X3, Y3}, {X4, Y4}, …として出力される。バଙ୍କチャ回路44は、そのうち、X2, X4

を削り（伝送しない）送信系列45, {U1(X1), V1(Y1)}, {U2(Y2), V2(X3)}, {U3(Y3), V3(Y4)}, …となる。このようにバଙ୍କチャにより4シンボルが3シンボルに変換されて出力される。符号化率は $R=4/6=2/3$ となる。

【0042】この場合は符号化率が $R=2/3$ の場合であるが、実際には伝送路の状況等に応じて設定される符号化率48によりバଙ୍କチャに必要なタイミングが符号化率設定回路47により作られ、バଙ୍କチャ回路44が操作される。

【0043】次に受信側の動作を図1を参照して説明する。予め本実施の形態のビタビ復号同期判定装置が用いられるシステムにおいて使用される複数の符号化率が符号化率設定回路1により設定されている。制御回路6は、設定された複数の符号化率から任意の符号化率を選択して、該符号化率に応じたデバଙ୍କチャ・タイミング信号18を出力するように符号化率選択回路2に指示する。

【0044】ここで受信信号列11{11a, 11b}を{U1, V1}, {U2, V2}, {U3, V3}, …とした場合、3シンボルを4シンボルに符号補填する操作をデバଙ୍କチャ回路3により行う。この場合、どのシンボルからの3シンボルでもよいわけではなく、この図1ではU1から戻した場合は通常に戻る。ただし、X2, X4はバଙ୍କチャにより失われているので、0などで埋められる。

【0045】ここでU2からデバଙ୍କチャを行なった場合は復号系列3a, 3bは、

X: Y2, 0, Y4, 0

Y: X3, Y3, X5, Y5

となり、正しくデバଙ୍କチャできない。U3からデバଙ୍କチャを行った場合も同様に正しくデバଙ୍କチャできないことになる。U4からの場合は3シンボルを1周期としているため、U1からデバଙ୍କチャを行った場合と同様となり、正しいデバଙ୍କチャがなされる。

【0046】そのため、符号化率 $R=2/3$ においては、自由度が3あるといい、一つの符号化率につき、3つの位置での誤りの状態を判定する必要がある。各符号化率における自由度は異なる。デバଙ୍କチャされた信号12はビタビ復号器4により復号され、もとの情報系列13が出力される。

【0047】誤り率検出回路5は、ある一定の期間（ビタビ復号が十分収束し、収束にかかる時間の割合がそのうち大きな部分を占めない程度に長く）の誤り率を判定し、所定の誤り率を上回るか否かを示す誤り率判定信号15を制御回路6に送る。

【0048】制御回路6は、所定の誤り率を上回る場合、符号化率が誤っていたかあるいはデバଙ୍କチャの位相が誤っていたかのいずれかであるので、現在選択されている符号化率でまだ検索していないバଙ୍କチャ位置が



あればバンクチャ位置をずらし、現在選択されている符号化率で全ての自由度を検索済みであれば新たな符号化率を選択するように、符号化率選択回路2に対して検索信号16を出力する。また、デバンクチャタイミング信号18を出力するように符号化率選択回路2に対して指示する。

【0049】符号化率選択回路2は、検索信号16がくると、現在選択されている符号化率でまだ検索していないバンクチャ位置があればバンクチャ位置をずらしたデバンクチャタイミング信号を出力し、現在選択されている符号化率で全ての自由度を検索済みであれば、設定された符号化率17から新たな符号化率を選択し、その符号化率に応じたデバンクチャタイミング信号18をデバンクチャ回路3に出力する。

【0050】このように、各設定された符号化率17についてのみ、それぞれの符号化率について、バンクチャ位置をずらしながら誤り率を判定し、その符号化率の自由度分だけ、例えば $R=2/3$ の場合は3回調べて、正しくないと判断された場合、次に設定された符号化率のデバンクチャをおこなうといったように順次調べていく。また、ひとつの符号化率について、2まわり $R=2/3$ の場合は6回といったように、複数回調べる方法もある。

【0051】次に、制御回路が制御する復号手順を図3のフローチャートに示す。同図において、まず符号化率 $R=1/2$ が設定されているか否かを調べる(ステップS10)。符号化率 $R=1/2$ が設定されていないければ(ステップS10の判定でNo)、符号化率 $R=1/2$ の処理を省略して、ステップS20へ移る。符号化率 $R=1/2$ が設定されていれば(ステップS10の判定でYes)、 $R=1/2$ のバンクチャ位置検出を行う(ステップS12)。次いで誤り率が所定の誤り率より小さいか否かを判定する(ステップS14)。誤り率が所定の誤り率より小さければ(ステップS14の判定でYes)、符号化率 $R=1/2$ が正しいので同期判定へ移る。誤り率が所定の誤り率に等しいか大きければ(ステップS14の判定でNo)、この符号化率 $R=1/2$ が誤りであるので、次に符号化率 $R=2/3$ が設定されているか否かを調べる(ステップS20)。

【0052】符号化率 $R=2/3$ が設定されていないければ(ステップS20の判定でNo)、符号化率 $R=2/3$ の処理を省略して、ステップS30へ移る。符号化率 $R=2/3$ が設定されていれば(ステップS20の判定でYes)、 $R=2/3$ のバンクチャ位置検出を行う(ステップS22)。次いで誤り率が所定の誤り率より小さいか否かを判定する(ステップS24)。誤り率が所定の誤り率より小さければ(ステップS24の判定でYes)、符号化率 $R=2/3$ が正しいので同期判定へ移る。誤り率が所定の誤り率に等しいか大きければ(ステップS24の判定でNo)、この符号化率 $R=2/3$

が誤りであるので、次に符号化率 $R=3/4$ が設定されているか否かを調べる(ステップS30)。

【0053】以下、同様に設定された符号化率について、次々に符号化率を変えながら、誤り率が所定の誤り率より小さくなるまで繰り返す。そして、符号化率 $R=8/9$ のバンクチャ位置検出(ステップS82)を行った後の誤り率判定(ステップS84)において、誤り率が所定の誤り率より小さければ(ステップS84の判定でYes)、この符号化率 $R=8/9$ が正しいものとして同期判定へ移る。

【0054】符号化率 $R=8/9$ が設定されていないか(ステップS80の判定でNo)、誤り率が所定の誤り率に等しいか大きければ(ステップS84の判定でNo)、この符号化率 $R=8/9$ が誤りであるので、最初に戻る。

【0055】ここで従来の手順である図11のフローチャートと比較した場合、各符号化率について設定されているか否かを判定し、設定されていない符号化率については、その符号化率のバンクチャ位置検出を省略しているため、高速に同期判定が可能となる。

【0056】例えば伝送路の規定により、符号化率 $R=1/2$ 、 $2/3$ 、 $5/6$ しか使用しない時、自由度の合計は10となり、符号化率 $R=1/2$ ないし $8/9$ のすべての符号化率を調べる場合の検索数34に対し、 $1/3$ 以下となっている。

【0057】また同期引込み検索は、各符号化率は、各自由度だけでなく、その他に、複数の位相状態を検索する場合(位相不確定性サーチ)でも、当然使用できる。その場合はさらに同期引き込み時間の短縮効果は高い。

【0058】図1における誤り率検出回路7の構成方法にはいくつかの方法があり、ビタビ復号のバスメトリックの大きさを使用するもの、正規化処理の大きさ、頻度を利用するもの等が考えられる。これらを使って誤り状態を判定し、また同期判定、非同期判定のしきい値を別にすることでヒステリシスな特性を持たせ、誤り同期を防ぎ、簡単に同期が外れないようにして、同期性能を上げることも可能である。

【0059】上記の誤り率検出に正規化処理の大きさを利用する場合、図9に示した構成のビタビ復号器を用いるならば、最尤判定部905から正規化回路902へ送られる正規化データ $\Gamma_{min}$ を用いることができる。またACSユニットの構成法によっては、バスメトリックレジスタのオーバフローが見込まれる時点で正規化処理を行うことも可能であり、この場合には正規化処理が起きる頻度すなわち正規化頻度が誤り率の大きさを現すことになる。

【0060】次に、本発明に係るビタビ復号同期判定装置の第2の実施の形態の構成を図4のブロック図に示す。本第2の実施の形態は、硬判定の結果とビタビ復号一再符号化したものを比較して誤り状態を判定する構成

例である。

【0061】図4において、本実施の形態のビタビ復号同期判定装置は、符号化率設定回路1、符号化率選択回路2、デバウチン回路3、ビタビ復号器4、誤り率検出回路5、制御回路6、たたみ込み符号器7、一致検出回路8、簡易判定回路9及び遅延回路10を備えて構成される。なお符号1から6までの構成要素は第1の実施の形態の構成要素とほぼ同様の構成要素である。

【0062】本実施の形態においては、デバウチン回路の出力が簡易判定回路9により簡易判定され、ビタビ復号及びたたみ込み符号器7による再符号化に要する時間に相当する時間だけ遅延回路10により遅延される。ビタビ復号された復号信号13は、送信側と同じ構成のたたみ込み符号器7により再符号化され、再符号信号20と遅延回路10の遅延出力信号22とが一致検出回路8により一致検出される。再符号信号20と遅延出力信号22との不一致が検出される毎に誤り信号23が一致検出回路8から誤り率検出回路5へ送出される。誤り率検出回路5は、所定時間内の誤り信号23を計数し、所定の誤り率と比較して誤り率を判定し制御回路6へ通知する。以後の動作は、第1の実施の形態と同様であるので省略する。

【0063】

【発明の効果】以上説明した様に本発明によれば、可能な符号化率の内、使用する符号化率の種類が少ない時には、使用・不使用を各符号化率ごとに設定できるように構成するので使用しない符号化率に対する復号を省略し同期引き込み検索時間を短くできるという効果がある。また、位相不確定性までを含めるとさらに効果は大きい。

【図面の簡単な説明】

【図1】本発明に係るビタビ復号同期判定装置の実施の形態を示すブロック図である。

【図2】符号化率設定回路の構成例を示す詳細回路図である。

【図3】図1のビタビ復号同期判定装置の動作手順を示すフローチャートである。

【図4】硬判定の結果とビタビ復号一再符号化したものを比較して誤り状態を判定する場合の構成例を示すブロック図である。

【図5】硬判定とビタビ復号による誤り訂正を説明する図である。

【図6】たたみ込み符号器の構成を示すブロック図である。

【図7】ビタビ復号法を説明するトレリス線図である。

【図8】ビタビ復号に於けるバスマトリック更新の説明図である。

【図9】ビタビ復号器の構成例を示すブロック図である。

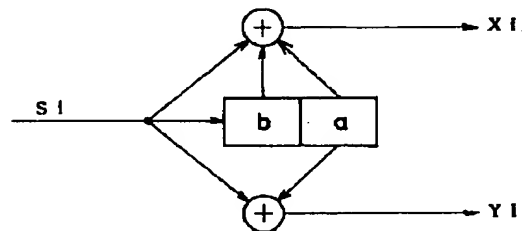
【図10】伝送路の状況に応じた符号化率で伝送するバウチン符号化装置の構成を説明するブロック図である。

【図11】従来のビタビ復号同期判定手順を示すフローチャートである。

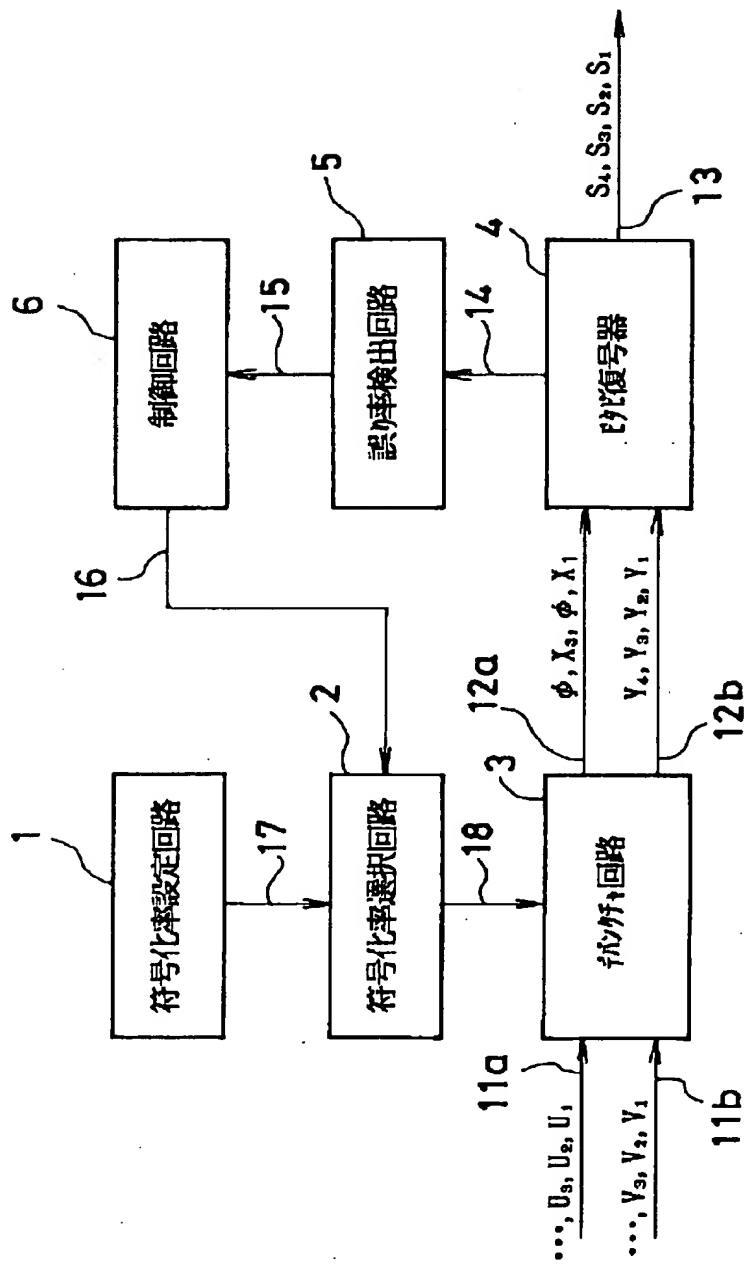
【符号の説明】

- |                    |            |           |
|--------------------|------------|-----------|
| 1 符号化率設定回路         | 2 符号化率選択回路 | 3         |
| デバウチン回路            |            |           |
| 4 ビタビ復号器           | 5 誤り率検出回路  | 6 制御回路    |
| 11 a, 1 b QPSK復号信号 | 12 a, 12 b | デバウチン信号   |
| 13                 | ビタビ復号信号    |           |
| 14                 | 15 誤り率判定信号 | 16        |
| 7 符号化率信号           | 18         | デバウチン制御信号 |

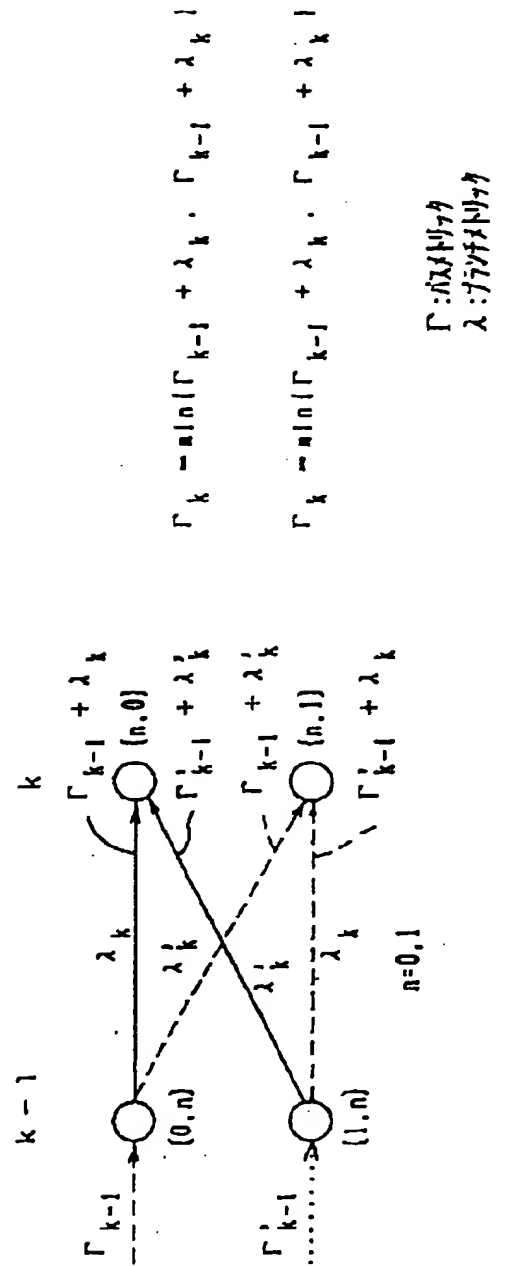
【図6】



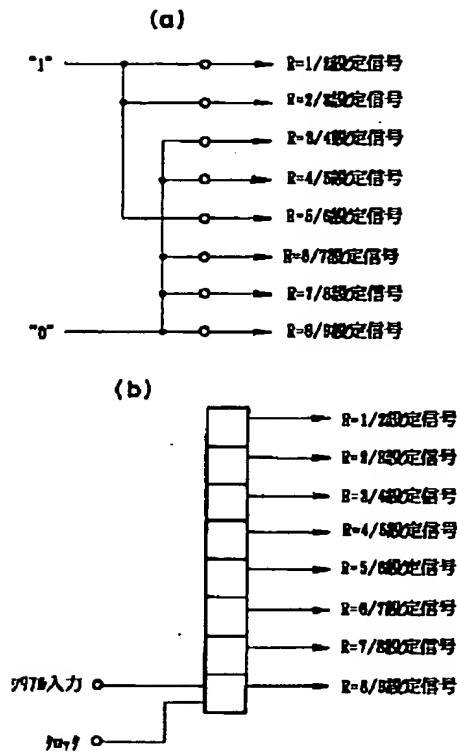
【図1】



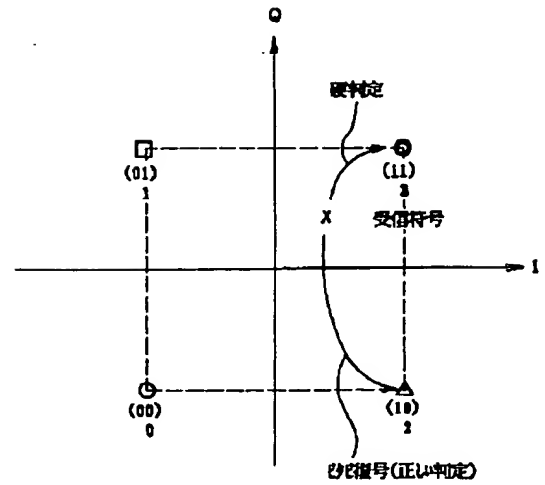
【図8】



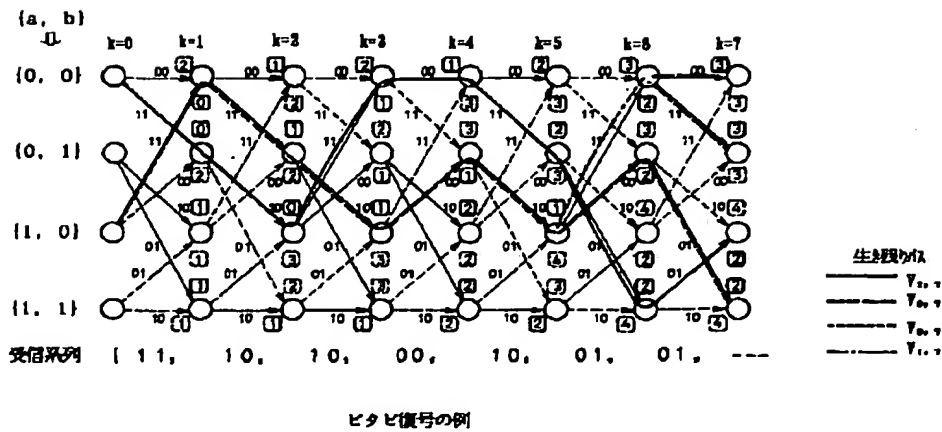
【図2】



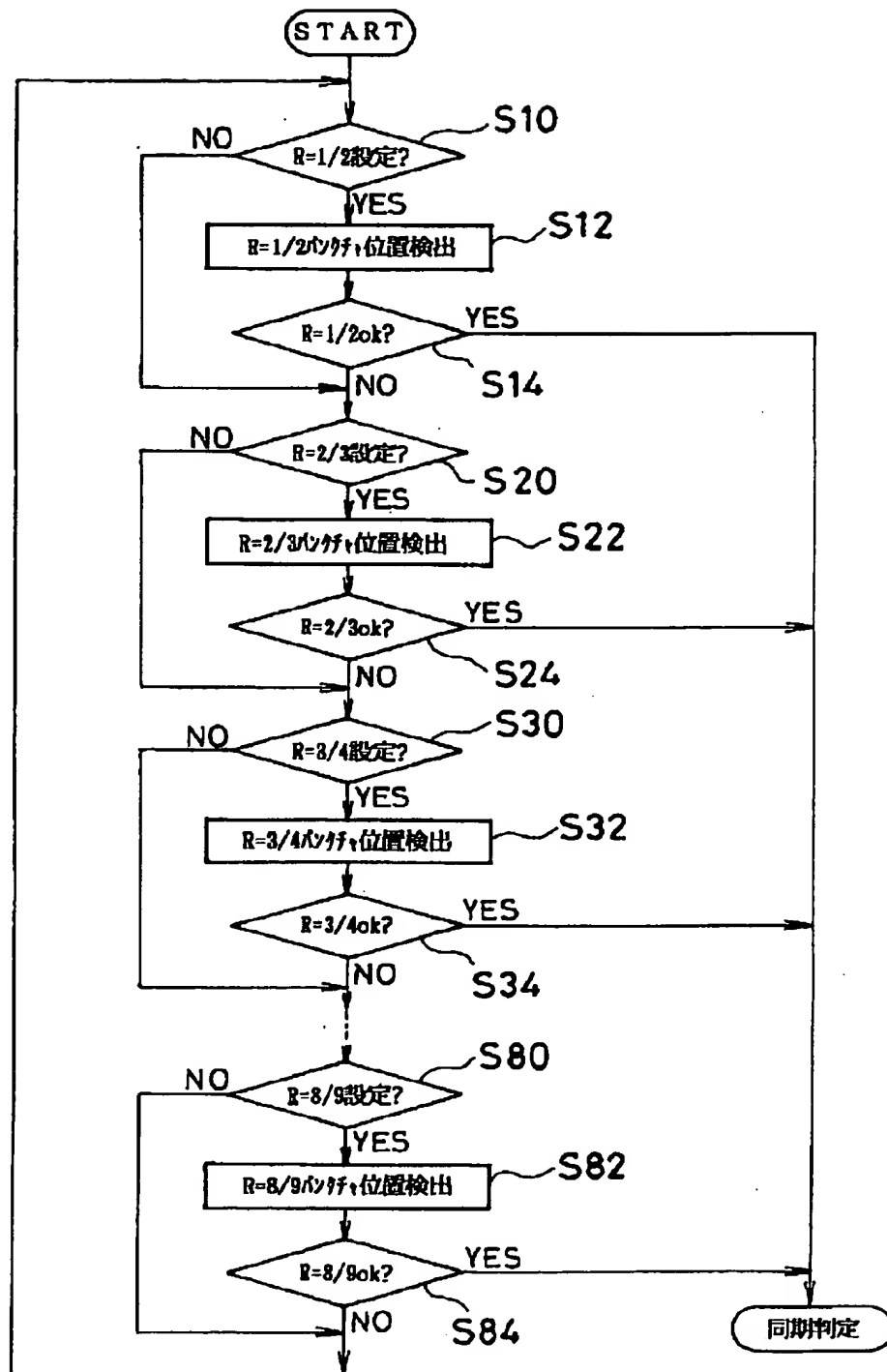
【図5】



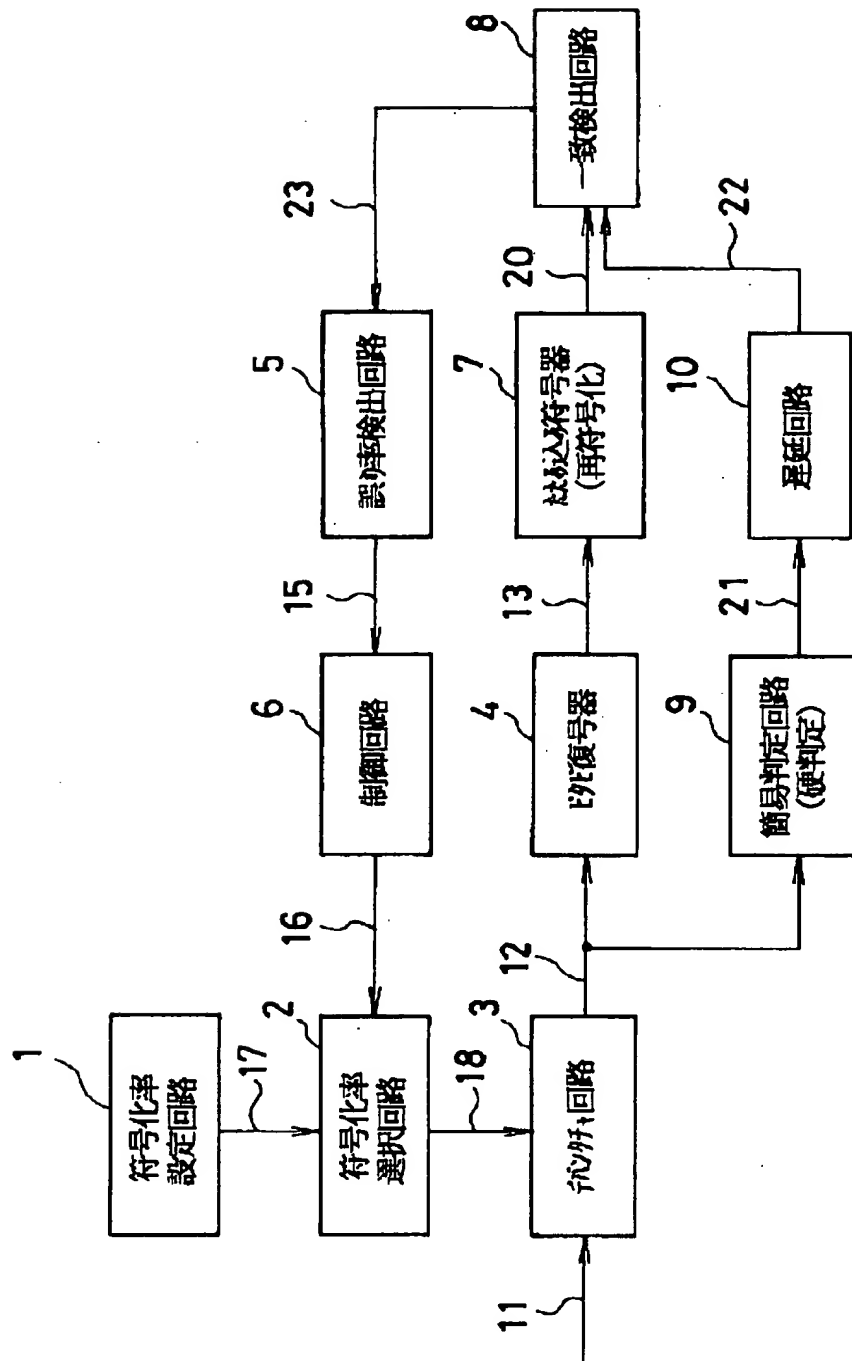
【図7】



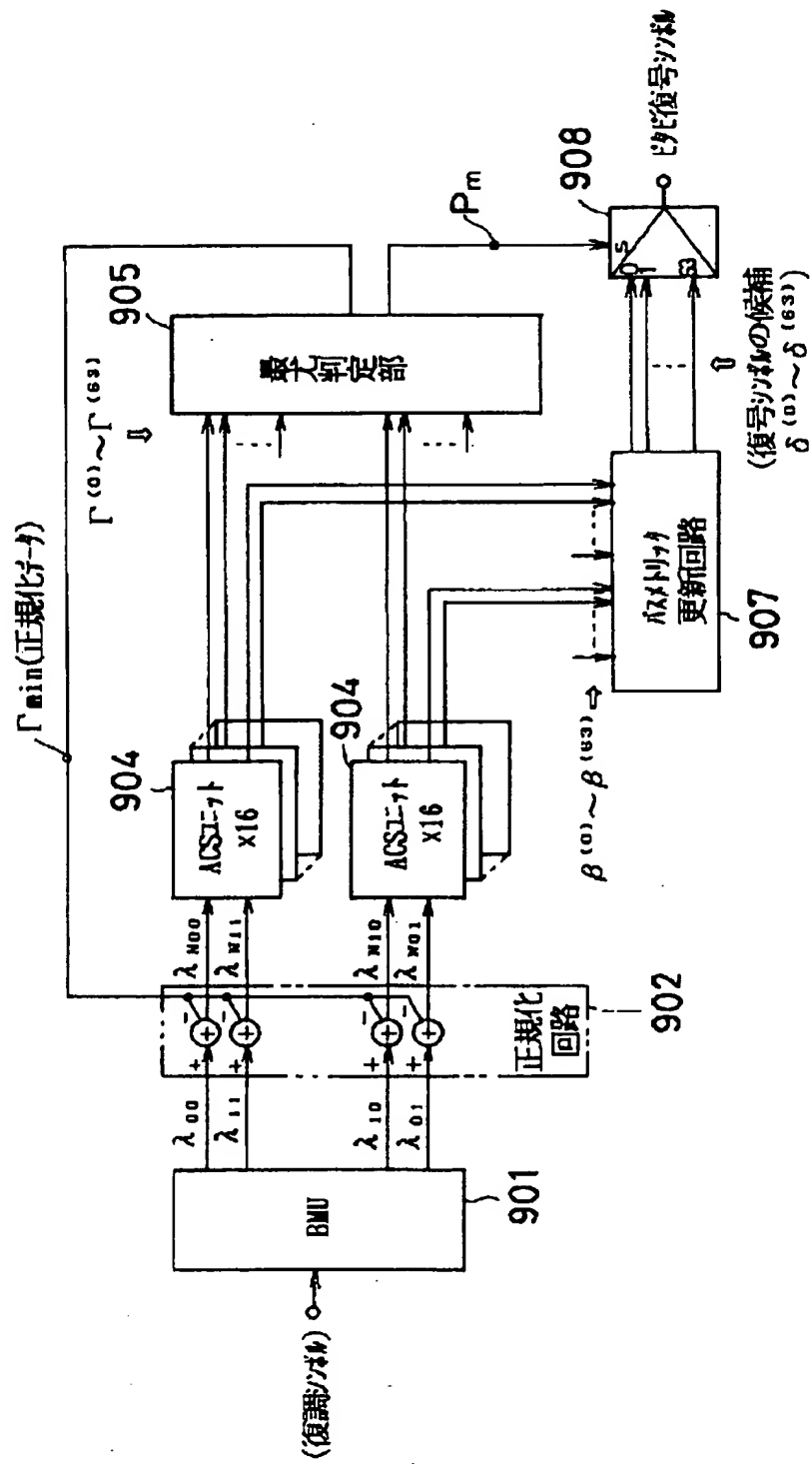
【図3】



【図4】

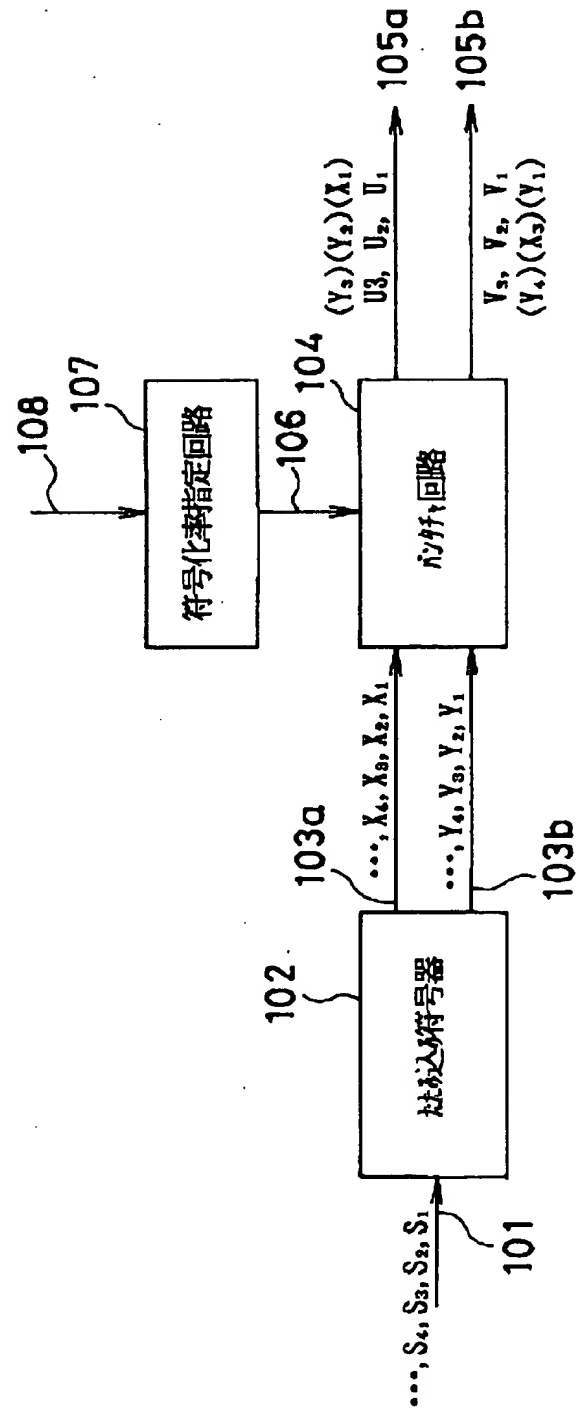


【図9】



ピタゴラス候補の構成例(状態数: NS=64, 拘束長: L=7)

【図10】





【図11】

